Universal Reference Drill-Down

David Turanski

dturansk@eos.hitc.com

31 October 1995

Overview

Universal Reference (UR)

- A persistent, transportable reference to ECS objects (e.g., data and services)
- Provides location transparency and protocol independence
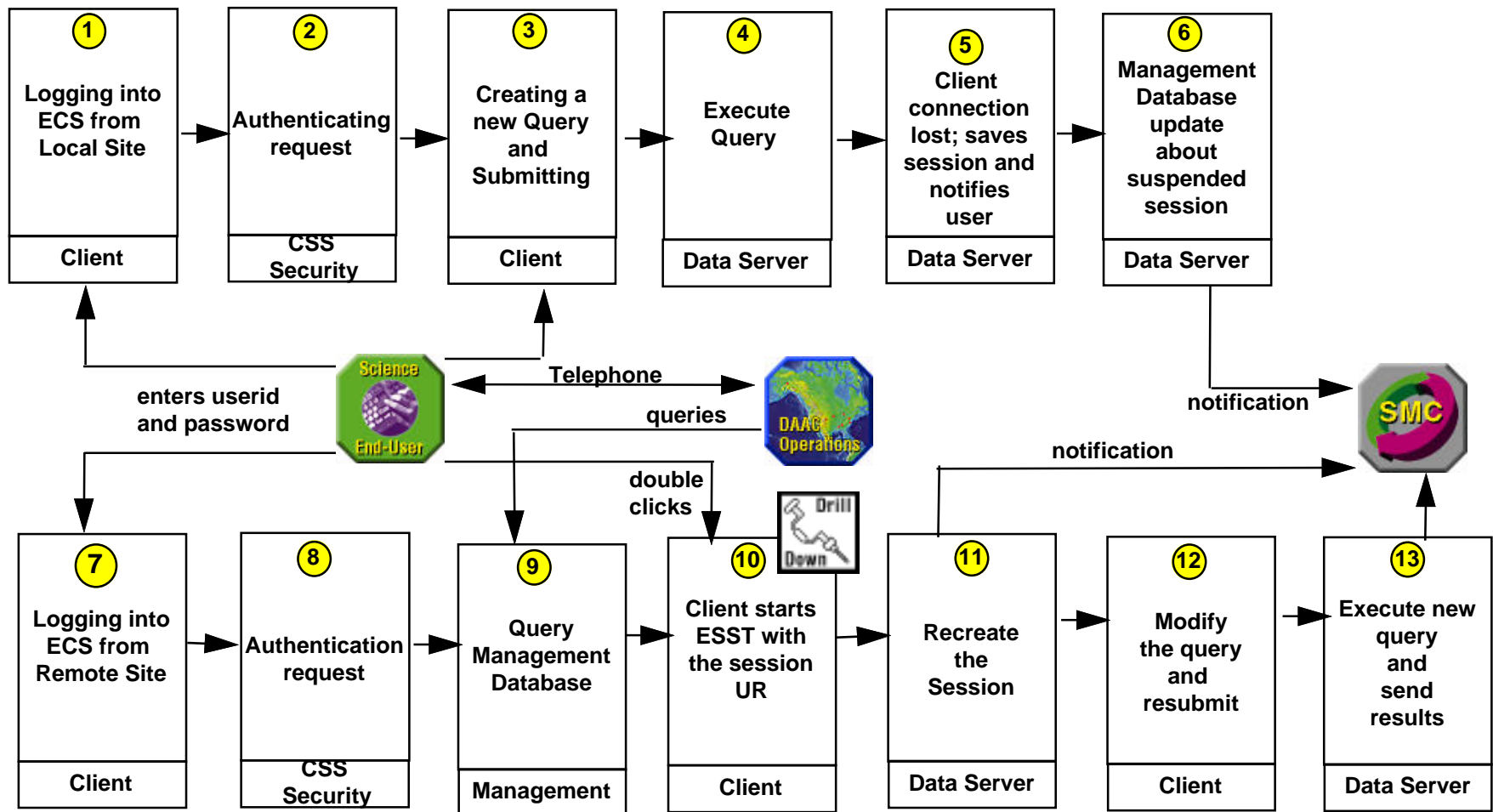
ECS Context

- Used to save and restore data, services, sessions, subscriptions, and requests

Scenario Context

- Resumption of disconnected session: User resumes saved session from a UR
- Used implicitly in majority of ECS scenarios

Disconnected Session
Functional Flow Diagram

①	②	③	④	⑤	⑥
Logging into ECS from Local Site	**Authenticating request**	**Creating a new Query and Submitting**	**Execute Query**	**Client connection lost; saves session and notifies user**	**Management Database update about suspended session**
Client	CSS Security	Client	Data Server	Data Server	Data Server

Science End-User

Telephone

DAAC Operations

enters userid and password

queries

notification

double clicks

notification

Drill Down

SMC

⑦	⑧	⑨	⑩	⑪	⑫	⑬
Logging into ECS from Remote Site	**Authentication request**	**Query Management Database**	**Client starts ESST with the session UR**	**Recreate the Session**	**Modify the query and resubmit**	**Execute new query and send results**
Client	CSS Security	Management	Client	Data Server	Client	Data Server

UR Design Features

More than a URL

- **A key component of the ECS middleware framework which supports the addition of new data types and services**

- **Provides a common mechanism for users and software components to save, restore, and transport objects**
  - **State of objects may be preserved**

- **Provides location transparency**
  - **The network location of UR and referenced objects is transparent to ECS applications**

- **Provides protocol independence**
  - **Encapsulation ensures design is not impacted by architectural transitions (e.g., DCE to CORBA)**

- **Supports nested references**
  - **URs can be linked together, e.g., a distributed query contains a collection of URs for each sub-query**
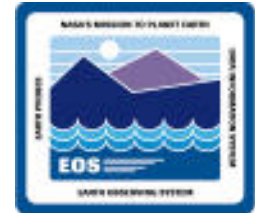
Software Design

There are different sub-types of URs. Each UR refers to an object (a.k.a. "Provider") which knows how to reconstitute itself from the UR
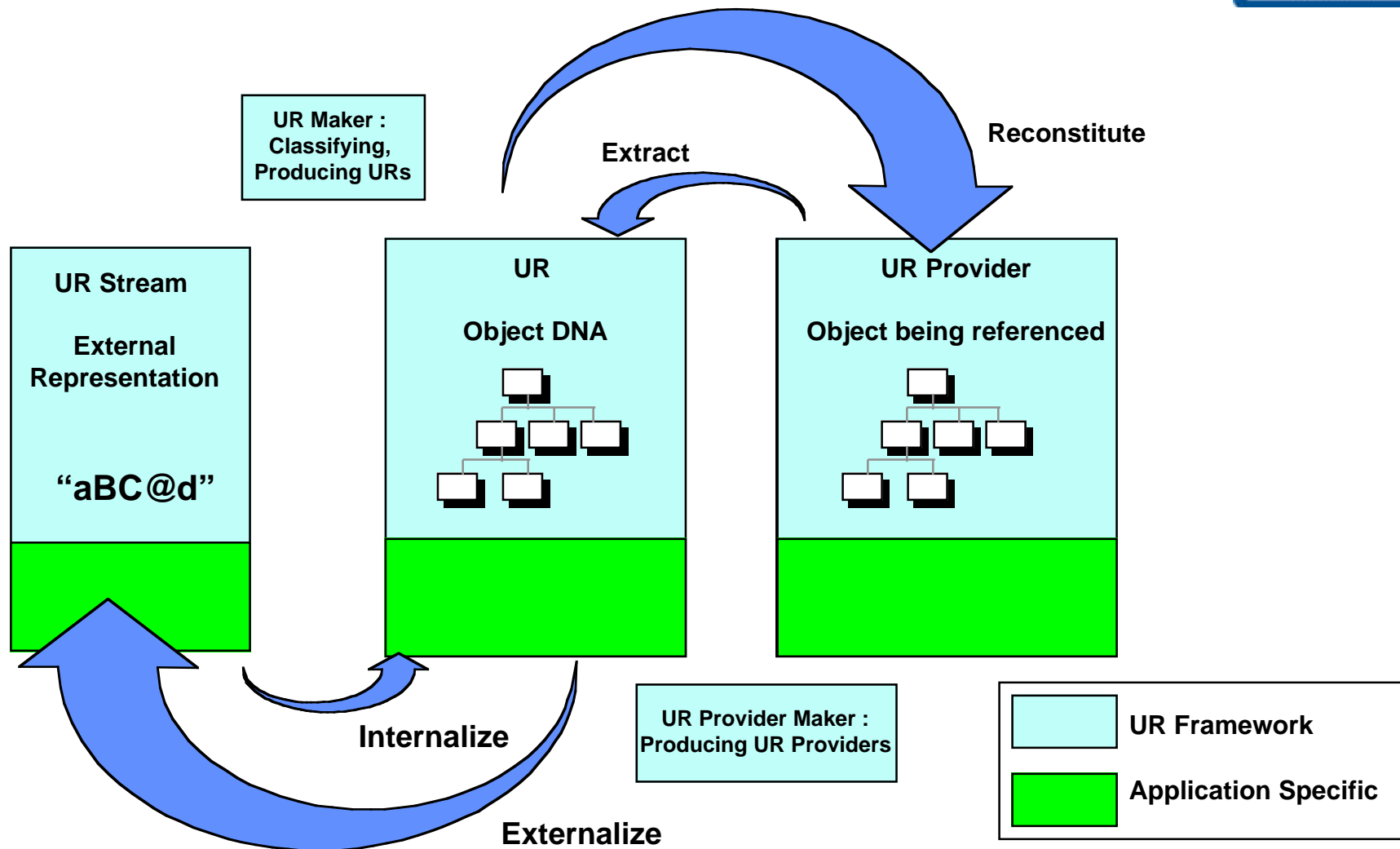
ECS UR providers include:

- Data Granule
- Advertisement
- Subscription
- Session
- Request

Helper classes are provided to allow ECS applications to reconstitute an object from its UR without knowing the UR's type in advance

- URMaker - creates a UR object from its external representation (file)
- URProviderMaker - creates a provider object from a UR object

UR Solution Overview

UR Maker :
Classifying,
Producing URs

Extract

Reconstitute

UR Stream

**External
Representation**

"aBC@d"

UR

Object DNA

UR Provider

Object being referenced

Internalize

UR Provider Maker :
Producing UR Providers

Externalize

UR Framework

Application Specific

Software Design

Example: Resumption of disconnected session

- User activates desktop icon representing a disconnected session (in a suspended state)

- The icon is associated with a file containing an external representation of a UR

- The desktop requests the URMaker to create a UR object

- The desktop determines from the UR type which client tool to launch

- The client tool requests the URProviderMaker to create a provider object (the saved session) from the UR

- The client tool asks the provider object to restore the session's state using data contained in the UR

Resumption of Disconnected Session Event Trace
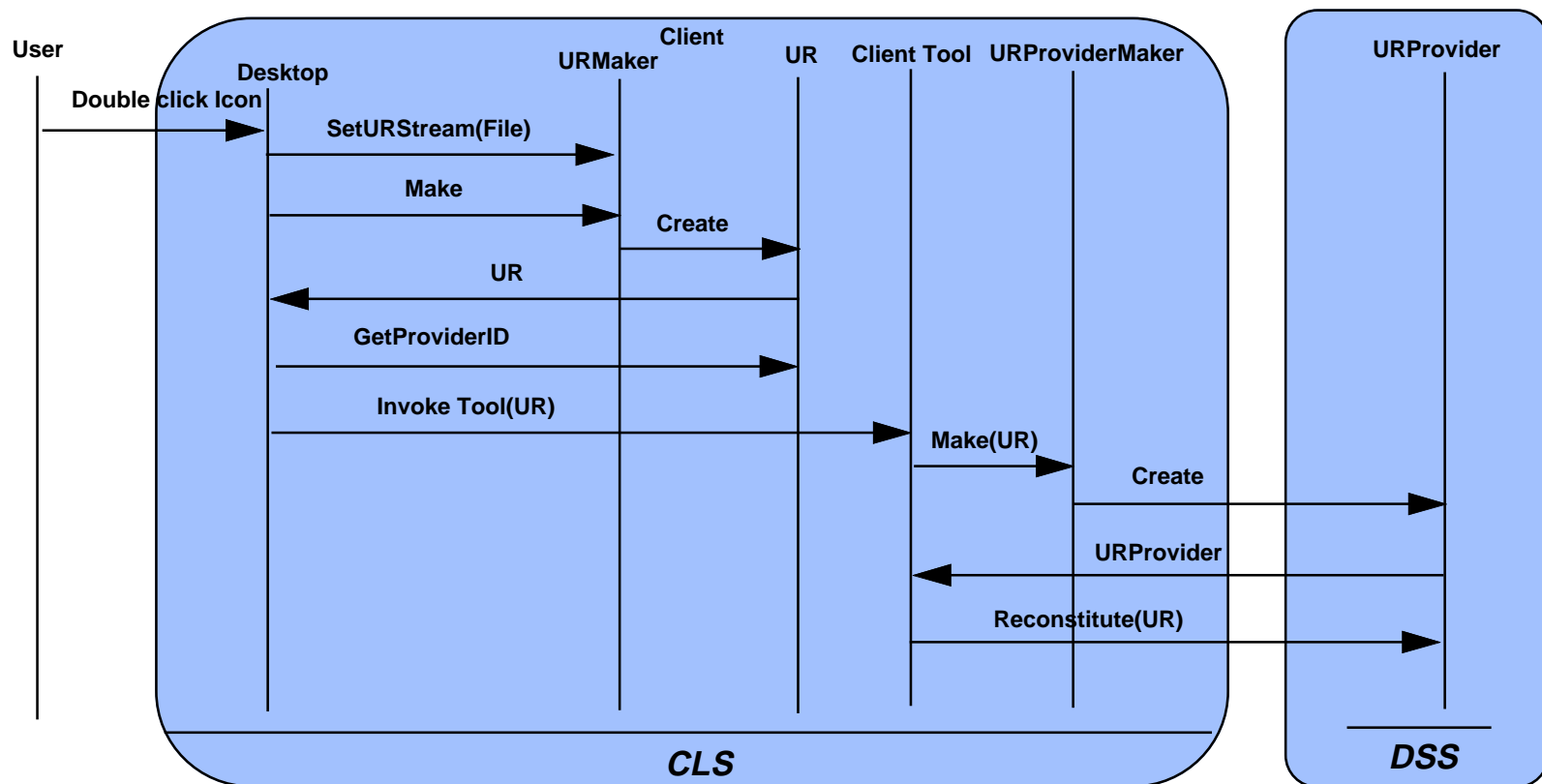
For details, reference:
DID 305-CD-020-001 SDPS/CSMS Design Overview

Summary

As a key component of ECS middleware, the UR Provides:

- **Persistent reference to ECS objects**

- **Location and protocol independence**

- **An extensible Object-Oriented Framework for ECS service providers to implement new services**

- **A consistent application interface**

- **A human-readable and transportable external representation**